Qu'est-ce qu'Astrabase ?

Guide public complet - edition 18 octobre 2026 - Astrabase V2

01 Comprendre Astrabase

Astrabase est une plateforme de creation et de gestion de projets web. Son ambition est de reunir la creation visuelle, le deploiement, l'hebergement de sites statiques, les outils de base de donnees, l'authentification, le stockage, l'observabilite, le referencement et un ecosysteme pour les createurs de logiciels. Cette ambition ne signifie pas que tous les modules ont la meme maturite. Ce guide distingue trois statuts : DISPONIBLE signifie qu'une fonction existe dans le produit ; SOUS CONDITIONS signifie qu'elle depend d'autorisations, d'une configuration ou d'un pilote ; EN PREPARATION signifie qu'elle ne doit pas etre consideree comme un service commercial deja livre. Les details peuvent evoluer ; verifier la console et les conditions de son compte. Adresses principales : https://astrabase.fr/ pour la presentation, https://cloud.astrabase.fr/ pour la console et https://community.astrabase.fr/ pour la communaute. Le domaine https://api.astrabase.fr/ est reserve aux usages API compatibles avec les fonctions activees.

02 Pour qui, et pour quoi faire ?

Astrabase vise notamment les developpeurs independants, les createurs de SaaS, les petites equipes, les freelances et les personnes souhaitant creer un site sans coder. Un createur peut monter un site multipages avec Site Studio, un developpeur peut publier une compilation React/Vite ou Astro, et une equipe peut gerer des donnees et des acces dans une console. Un site vitrine statique constitue un cas d'usage simple. Une application SaaS avec logique dynamique specifique peut exiger une API externe, une configuration avancee ou une infrastructure dediee. Astrabase ne promet pas l'hebergement de n'importe quelle application ni une equivalence complete avec des plateformes tierces.

03 Panorama des fonctions

- DISPONIBLE : Site Studio pour sites statiques multipages ; editeur visuel, themes, sections, medias et apercu mobile.
- DISPONIBLE : publication de contenus HTML statiques et de compilations adaptees React/Vite ou Astro.

- DISPONIBLE : gestion de projets, versions de deploiement, domaines, diagnostics et historiques selon la configuration.

- DISPONIBLE ou SOUS CONDITIONS : PostgreSQL, tables SQL, API, Auth, Storage et divers services cloud.

- SOUS CONDITIONS : fonctions backend, Realtime, automatisations et hebergement dynamique lorsque explicitement autorises.

- EN PREPARATION ou PILOTE : certaines integrations avancees, la couverture totale du SDK et des capacites serveur securisees a grande echelle.

Les intitulés de modules dans la console indiquent une surface de produit, pas une garantie que chaque fonction soit ouverte a tous les comptes.

04 Demarrage : compte, projets et collaborateurs

Ouvrir la console, creer un compte puis consulter son tableau de bord. Les projets disposent de droits et de parametres propres. Les equipes peuvent regrouper plusieurs projets et membres lorsque le plan et les controles d'acces le permettent. Avant d'inviter un collaborateur, definir ses besoins de lecture ou d'ecriture. Donner seulement les permissions necessaires et verifier les droits de chaque membre apres modification. Ne pas partager ses identifiants personnels.

05 Site Studio : creer sans coder

Site Studio est un editeur visuel pour sites statiques multipages. Il permet de choisir un modele, modifier les textes, sections, liens et elements visuels, rearranger la page, gerer des images, consulter un apercu bureau ou mobile et preparer une publication. Un site statique peut presenter une entreprise, une association, un portfolio ou une offre. Il ne devient pas automatiquement une application avec base de donnees, paiement, espace membre et logique serveur. L'editeur et les fonctions d'hebergement ont des limites distinctes. Parcours conseille : creer un projet Static / HTML, choisir un modele, modifier les pages, inspecter la version mobile, verifier les liens, publier apres confirmation et conserver une version precedente utilisable pour un retour arriere.

06 Hebergement et deploiement : ZIP, Git, URL

Astrabase peut publier un site HTML ou une sortie de build statique. Avec React/Vite et Astro, la publication par archive ZIP doit contenir les fichiers compiles, avec index.html a la racine de l'archive publiee. Les sources seules ne remplacent pas une compilation. Les depots Git et certains parcours de deploiement existent dans le produit ; le comportement des depots prives, builds et previews depend des possibilites effectivement activees. Les builds de code tiers sont soumis a des restrictions de securite pendant la beta. Les domaines personnalises et HTTPS passent par la configuration DNS et le reverse proxy. La propagation DNS n'est pas instantanee. Un rollback doit etre envisage avant toute mise a jour critique.

07 Ce que Next.js ne signifie pas sur Astrabase

Un site React exporte en fichiers statiques ne requiert pas le meme environnement qu'une application Next.js dynamique avec execution serveur. L'hebergement dynamique Next.js est un parcours pilote sous conditions, non une promesse d'execution universelle. Verifier les exigences : serveur Node, traitements d'arriere-plan, base de donnees, sessions, stockage persistant et ressources. Si le projet necessite l'execution de code non fiable, ne pas considerer le VPS partage comme une isolation suffisante.

08 Backend : PostgreSQL et isolation logique

Astrabase s'appuie sur PostgreSQL. Les projets peuvent utiliser des schemas et roles SQL distincts. Cette separation est logique ; elle n'equivaut pas a une machine physique independante ou a une garantie absolue entre applications. Des tables, requetes et requetes API peuvent etre administrees selon les roles et les fonctions activees. Toujours configurer les autorisations des donnees avant d'ajouter des utilisateurs finaux ou d'exposer une API. PostgreSQL, pgvector et les interfaces SQL peuvent etre presentes. La presence de pgvector ne signifie pas que les recherches vectorielles, l'indexation de documents et toutes les pipelines IA sont preconfigurees.

09 RLS, policies et clefs d'API

RLS (Row-Level Security) est un controle d'acces aux lignes PostgreSQL. Activer RLS seul ne cree pas de regles d'autorisation : sans policy adaptee, les usages peuvent etre bloques ou presenter des comportements differents selon le role. Les protections dependront aussi des roles SQL, du proprietaire des tables, des privileges serveur et des chemins d'acces API. Faire verifier les policies, tester la lecture/ecriture d'un utilisateur non autorise et ne jamais utiliser une clef d'administration dans un navigateur. Les audits precedents ont observe des tables tests avec RLS active sans policy, FORCE RLS non active. Il serait faux de conclure que toute la protection multi-tenant est complete. Ces mesures restent a controler projet par projet.

10 Auth : identite et sessions

L'authentification gere selon les parcours disponibles l'inscription, la connexion, les sessions et certains mecanismes de verification. Les capacites MFA, passkeys et OAuth doivent etre verifiees dans la version de console utilisee ; ne pas supposer qu'elles couvrent tous les ecrans et integrations. Les utilisateurs doivent disposer de secrets distincts, de sessions revocables et de controles adaptes a leur niveau de risque. Les administrateurs devraient proteger les comptes sensibles avec des facteurs supplementaires lorsqu'ils sont disponibles.

11 Storage : fichiers, buckets et droits

Storage permet, selon les fonctions activees, de gerer des espaces de fichiers associes aux projets, d'envoyer et de consulter des objets et de definir des regles d'acces. Ne stocker aucun secret dans un fichier public. Avant d'exposer des fichiers utilisateurs, verifier les permissions, les tailles maximales, les types MIME, les liens publics ou prives et la politique de suppression. Sauvegarder les objets critiques en dehors de leur repertoire applicatif.

12 Fonctions, Realtime, Cron et webhooks

Les modules Functions, Realtime, taches planifiees et webhooks couvrent des besoins differents : execution d'une action, diffusion d'un evenement, planification d'une operation et reception d'un message externe. Chaque module peut demander une activation ou un parametrage specifique. Les fonctions qui executent du code doivent etre traitees comme sensibles. Tant que les runners partagent le socket Docker de l'hote, l'execution de code de tiers non fiables ne doit pas etre ouverte. Le fait qu'un bouton existe ne remplace pas une autorisation serveur. Pour les webhooks, utiliser une signature validee, des traitements idempotents et un journal de suivi. Pour les crons, surveiller les executions, limiter les privileges et prevoir la reprise apres une erreur.

13 Secrets, roles et controles d'acces

Les secrets d'API, mots de passe, jetons et identifiants de services doivent rester hors des depots publics, des pages HTML et des captures. Utiliser un mecanisme de variables et secrets cote serveur quand il est disponible. Appliquer le principe du moindre privilege : droits minimes pour chaque utilisateur, jeton et service. Un administrateur de projet ne devrait pas reutiliser ses clefs partout. Revoquer rapidement un acces devenu inutile.

14 SEO Console et referencement

SEO Console et les guides publics aident a examiner titres, descriptions, H1, canonical, robots.txt, sitemap et d'autres signaux de publication selon les modules disponibles. Le controle d'un referencement ne garantit jamais une position sur Google. Si Google Search Console ou un service externe est connecte, confirmer les autorisations et la bonne propriete. Une demande d'indexation et une page effectivement indexee sont deux situations differentes. Site Studio permet de travailler la structure de contenu, mais les fonctionnalites SEO par page et les automatismes doivent etre verifies dans la version activee.

15 Les quinze familles de services Astrabase

Servers : informations et operations relatives aux serveurs lorsque la connexion est autorisee. Mail : configuration et suivi d'envois selon SMTP et domaine ; la delivrabilite ne peut etre promise sans service actif. Git : depots, activite et operations de publication dans les limites des autorisations. Cache : outils de mise en cache de projet quand deployes. Buckets : stockage des objets et regles d'acces associees. DNS : verification de domaines, enregistrements et diagnostics. Monitor : disponibilite, historiques et observations selon sondes configurees. Automate : taches planifiees, webhooks et automatisations disponibles. Secrets : stockage et rotation de valeurs sensibles selon les mecanismes proposes. Analytics : metriques agregees et vues trafic selon collecte configuree. Errors : signalement, exploration et diagnostic d'erreurs. Flags : activation progressive de fonctionnalites. Builder : outils de generation et publication de pages simples. AI : assistance et fonctions IA selon configuration, credits et capacites activees. API Gateway : routes et acces aux services selon projets et permissions. Un service affiche dans un catalogue n'est pas automatiquement une infrastructure mature comparable a une offre autonome de grand fournisseur.

16 Journaux, analytics et diagnostic

Un tableau de bord peut afficher les versions deployees, les erreurs, des metriques, les releves d'audit et les usages agreges. Toujours distinguer demonstrateur visuel, metriques locales et mesure reelle d'un service en production. Les informations sensibles ne doivent pas etre ecrites dans les logs ni exposees au client. Revoir regulierement les echecs de deploiement, les permissions et les alertes ; ces outils facilitent l'exploitation mais n'eliminent pas les incidents.

17 Integrations, API et MCP

Une API REST et des parcours de connexion MCP sont proposes avec permissions adaptees. La connexion a ChatGPT depend de l'activation du connecteur et des autorisations accordees projet par projet. Les droits de modification sont differents des droits de consultation. Avant d'utiliser un SDK, une CLI ou une integration externe, verifier sa documentation, sa version et ses limites. Ne pas considerer comme stable une methode non publiee ou en preparation.

18 Plans, prix et quotas

La Landing presente des formules Free, Pro et Team. Les montants et quotas affiches peuvent etre amenes a evoluer et doivent etre recoupes avec l'ecran de facturation, les CGV et les limites effectivement appliquees. Les depassements eventuels, la facturation et les fonctions conditionnelles sont a lire avant un abonnement. Les ressources comme DB, Storage, Auth, bande passante, builds et fonctions doivent etre mesurees selon la configuration reelle. L'absence d'un quota public clair n'autorise pas a supposer un usage illimite.

19 Securite de la plateforme : protections presentes et risques

L'acces public utilise notamment Caddy et HTTPS. Le code inclut controles d'acces, mecanismes de limitation, restrictions de privileges et protections de certains webhooks. Ces protections reduisent des risques, sans constituer une certification de securite totale. Astrabase opere actuellement sur un VPS OVHcloud partage avec d'autres services. Les runners de build, de fonctions et de Next.js utilisent encore le socket Docker de l'hote : ce privilege constitue un risque residuel critique. Les restrictions de conteneurs et de reseau ne constituent pas une isolation prouvee contre un code hostile. L'ouverture de code non fiable a des tiers doit attendre une execution sur travailleurs reellement isoles ou une architecture appropriee. Aucun label ISO 27001, SOC 2 ou aucune region d'hebergement garantie ne doit etre suppose sans justificatif.

20 Sauvegardes, restauration et incidents

Des sauvegardes locales existent et un processus de verification de certaines generations a ete mis en place. Une copie chiffree hors VPS Restic/S3 et une restauration complete verifiee n'etaient pas etablies au moment de cette edition. Une sauvegarde sur le meme VPS ne protege pas contre la perte de toute la machine. Avant une utilisation critique, organiser des copies hors site, tester des restaurations, documenter les delais de reprise et attribuer les responsables d'incidents. En cas d'incident, interrompre si necessaire les acces a risque, preserver les journaux, examiner les droits affectes, restaurer de facon controlee et informer les personnes concernees selon les obligations applicables.

21 Protection des donnees et RGPD

Limiter la collecte de donnees personnelles, definir une base legale et une finalite, afficher des informations de confidentialite, respecter les droits d'acces et de suppression et choisir des durees de retention proportionnees. Pour une application cliente, le responsable de traitement et les sous-traitants ont des obligations contractuelles distinctes. Une offre cloud ne dispense pas de securiser les donnees propres a l'application. Lire les mentions legales, la politique de confidentialite, les CGU et les CGV sur https://astrabase.fr/. Verifier la region effective et les sous-traitants avant d'affirmer une localisation ou conformite particuliere.

22 Guide pratique : premier deploiement statique

1. Creer un compte et un projet adapte aux sites statiques.
2. Creer un site HTML ou compiler un projet React/Vite ou Astro localement.
3. Verifier que l'archive de publication contient index.html a sa racine.
4. Publier la compilation via le parcours ZIP ou Git effectivement disponible.
5. Tester la page publique, les liens et l'affichage mobile.
6. Configurer eventuellement le domaine DNS et attendre sa propagation.
7. Conserver la reference de la derniere version fonctionnelle pour rollback.

23 Guide pratique : premier projet avec donnees

1. Definir les tables et les donnees minimales de votre projet.
2. Creer le schema ou les ressources accessibles dans PostgreSQL.
3. Limiter les roles et creer des policies RLS si elles sont necessaires.
4. Tester les requetes depuis un compte non administrateur.
5. Utiliser uniquement les clefs publiques autorisees dans le navigateur.
6. Sauvegarder les donnees et verifier la possibilite de restauration.
7. Mettre en production uniquement apres revue des chemins d'acces sensibles.

24 Questions frequentes

Puis-je creer un site sans developpeur ? Oui, pour les sites statiques compatibles avec Site Studio. Puis-je heberger une application React ? Oui, si son artefact est un site statique compile adapte au mode de publication. Puis-je executer un backend arbitraire ? Non, l'execution de code tiers est soumise a de fortes restrictions pendant la beta. Puis-je mettre des donnees sensibles ? Seulement apres verification des controles, responsabilites, politiques d'acces et sauvegardes adaptes au besoin. Astrabase et Astrabase Community sont-ils identiques ? Non. Community est un espace distinct de decouverte de logiciels et de leurs createurs. Le site promet-il une conformite ou securite absolue ? Non. Les limites restantes sont exposees dans ce document.

25 Ressources utiles et mise a jour

Site : https://astrabase.fr/ Catalogue et limites : https://astrabase.fr/fonctionnalites-astrabase.html Guide hebergement : https://astrabase.fr/hebergement-sites-react-astro.html Site Studio : https://astrabase.fr/creer-site-web-sans-code.html Console : https://cloud.astrabase.fr/ Community : https://community.astrabase.fr/ Edition de reference : 18 octobre 2026. Ce guide est informationnel et doit suivre les fonctionnalites reellement validees avant chaque revision.